**Day-15(15-11-2024)**

* commands

Carriage return \r – cursor returns to the start of the same line we are working

New line \n - cursor points to the start of the next line

\b – backspace the values

* int x=10, y=20;

printf(“%d”, x, y); ---- o/p: 10

* int a=b=c=10;

a=b=c=50;

leads to the compile error we can’t assign values like this is C

* **Command line arguments:**
* ./app arg1 arg2🡺 to pass the arguments
* Always the arguments are taking as a string for the main
* If we give any other datatype as a arguments we have to convert that into that datatype
* Command line arguments are in the main itself
* Main(int argc ,char \*argv[] or char \*\*argv, char \*env)

Number of arguments which are the arguments

* You can change the variables of the arguments
* Whenever we give the application to client we should not provide the debugging mode(-g) and can provide for our purpose
* Eg : ./app 1 2 3 4 5 🡺 count=6(it count the application also)
* You can give the path name as a arguments at runtime

./app home/ramya home2/balivada

* Write return or exit to not to print remaining lines below that statement
* Write a program to read the contents from the file and store them in the employee records using command line arguments to supply the database

Program for command line arguments:

/\*

Command line arguments

main(int argc,char \*argv[])

\*/

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

int main(int argc, char \*argv[])

{

int i;

if(argc<2)

{

fprintf(stderr,"\ndest/source is missing");

fprintf(stderr,"\n./app src dest\n\n");

return 1;

}

printf("\nNo of args = %d",argc);

for(i=0;i<argc;i++)

printf("\nargv[%d] = %s",i,argv[i]);

printf("\n\n");

return 0;

}

Program 1 :

/\*

Command line arguments

main(int argc,char \*argv[])

\*/

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#define BUFF 1024

#define OR ||

int main(int argc, char \*argv[])

{

FILE \*fp;

// int i;

char fileName[BUFF];

char lines[BUFF];

// char \*tokens=NULL;

if((argc<2)OR(argc>2))

{

fprintf(stderr,"\nconfigFile is missing");

fprintf(stderr,"\n./app configFile.csv\n\n");

return 1;

}

strcpy(fileName,argv[1]);

fp = fopen(fileName,"r");

if(fp == NULL)

{

perror("fopen: ");

return 1;

}

memset(lines,'\0',BUFF);

while(fgets(lines,BUFF,fp)!=NULL)

{

lines[strlen(lines)-1] = '\0';

puts(lines);

memset(lines,'\0',BUFF);

}

fclose(fp);

printf("\n\n");

return 0;

}